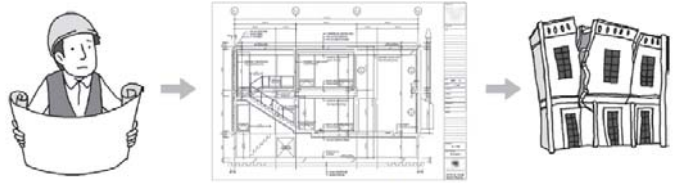
정규화
Chapter 07

- 이상현상
- 함수 종속성
- 정규형
- 정규화 다시 보기

이상현상

관계 스키마에 대한 좋은 설계



이상현상(anomaly)

- 관계 스키마가 잘못 설계되었을 때 나타날 수 있는 부작용
  - 수정 이상(update anomaly)
    - 데이터 수정(update)시, 중복된 데이터의 일부만 수정되어 데이터의 불일치 문제가 발생하는 현상
  - 삽입 이상(insertion anomaly)
    - 투플 삽입(insert)시, 불필요한 데이터를 함께 저장해야만 투플 삽입이 가능한 현상
  - 삭제 이상(deletion anomaly)
    - 투플 삭제(delete)시, 다른 필요한 정보까지 함께 삭제되는 현상

이상현상(anomaly) : 예

- 예) 학생수강 릴레이션
  - 키 : (학생번호, 강좌이름)

학생번호	학생이름	학과	주소	강좌이름	강의실
501	박지성	컴퓨터과	영국 맨체스타	데이터베이스	공학관 110
401	김연아	체육학과	대한민국 서울	데이터베이스	공학관 110
402	장미란	체육학과	대한민국 강원도	스포츠경영학	체육관 103
502	추신수	컴퓨터과	미국 클리블랜드	자료구조	공학관 111
501	박지성	컴퓨터과	영국 맨체스타	자료구조	공학관 111

이상현상(anomaly) : 예

- 수정 이상
  - 만일 어떤 학생의 주소가 수정될 때, 이 학생이 수강하는 일부 수강강좌 투플에서만 주소를 변경하면 데이터의 불일치 문제가 발생

  => 박지성 학생의 주소를 변경할 때, 자료구조 강좌 투플에 대한 주소만 변경하면, 박지성 학생 주소 데이터의 불일치 발생

학생번호	학생이름	학과	주소	강좌이름	강의실
501	박지성	컴퓨터과	영국 맨체스타	데이터베이스	공학관 110
401	김연아	체육학과	대한민국 서울	데이터베이스	공학관 110
402	장미란	체육학과	대한민국 강원도	스포츠경영학	체육관 103
502	추신수	컴퓨터과	미국 클리블랜드	자료구조	공학관 111
501	박지성	컴퓨터과	~~영국 맨체스타~~	자료구조	공학관 111

대한민국 수원

이상현상(anomaly) : 예

- 삽입 이상
  - 만일 신입 학생을 새로 삽입하려는데, 아직 수강 신청을 하지 않았다면 학생 정보를 입력할 수 없음

  => (수강강좌) 정보를 저장하는 것이 불가능하기에,
  불필요한 (가짜 수강강좌)정보를 강제로 넣어야 하는 것

학생번호	학생이름	학과	주소	강좌이름	강의실
501	박지성	컴퓨터과	영국 맨체스타	데이터베이스	공학관 110
401	김연아	체육학과	대한민국 서울	데이터베이스	공학관 110
402	장미란	체육학과	대한민국 강원도	스포츠경영학	체육관 103
502	추신수	컴퓨터과	미국 클리블랜드	자료구조	공학관 111
501	박지성	컴퓨터과	영국 맨체스타	자료구조	공학관 111
403	박세리	체육학과	대한민국 대전	가짜값	NULL

이상현상(anomaly) : 예

- 삭제 이상
  - 만일 수강강좌에 1명의 학생만이 신청한 상태인데, 이 학생에 관한 투플을 삭제하면 이 학생이 수강하는 강좌에 관한 정보도 릴레이션에서 함께 삭제됨

    => 예) 장미란 학생의 투플을 삭제하면, 스포츠경영학 강좌정보도 함께 삭제됨

학생번호	학생이름	학과	주소	강좌이름	강의실
501	박지성	컴퓨터과	영국 맨체스타	데이터베이스	공학관 110
401	김연아	체육학과	대한민국 서울	데이터베이스	공학관 110
~~402~~	~~장미란~~	~~체육학과~~	~~대한민국 강원도~~	~~스포츠경영학~~	~~체육관 103~~
502	추신수	컴퓨터과	미국 클리블랜드	자료구조	공학관 111
501	박지성	컴퓨터과	영국 맨체스타	자료구조	공학관 111

잘못 설계된 계절학기 수강 테이블

계절학기 수강 테이블 : Summer(sid, class, price)

SID	CLASS	PRICE
100	FORTRAN	20000
150	PASCAL	15000
200	C	10000
250	FORTRAN	20000

- 삭제 이상
  - 200번 학생의 계절학기 수강신청을 취소하면(투플을 삭제하면), C강좌의 수강료 정보도 함께 삭제됨

SID	CLASS	PRICE
100	FORTRAN	20000
150	PASCAL	15000
~~200~~	~~C~~	~~10000~~
250	FORTRAN	20000

잘못 설계된 계절학기 수강 테이블

- 삽입 이상
  - 계절학기에 자바 강좌를 개설하려고 하면(자바 강좌 투플을 삽입), 불필요한 가짜 학생 정보 SID를 강제로 넣어야 함

SID	CLASS	PRICE
100	FORTRAN	20000
150	PASCAL	15000
200	C	10000
250	FORTRAN	20000
가짜값	JAVA	25000

잘못 설계된 계절학기 수강 테이블

- 수정 이상
  - 학번이 100번인 학생에 대한 FORTRAN 강좌의 수강료를 20000원에서 15000원으로 수정하면, FORTRAN 강좌 수강료 데이터의 불일치 발생

SID	CLASS	PRICE
100	FORTRAN	~~20000~~ → 15000
150	PASCAL	15000
200	C	10000
250	FORTRAN	20000

수정된 (잘 설계된) 계절학기 수강 테이블

- 테이블의 구조를 수정하여 이상현상이 발생하지 않는 사례

Summer(sid, class, price)

sid	class	price
100	FORTRAN	20000
150	PASCAL	15000
200	C	10000
250	FORTRAN	20000

SummerPrice(class, price)

class	price
FORTRAN	20000
PASCAL	15000
C	10000

SummerEnroll(sid,class)

sid	class
100	FORTRAN
150	PASCAL
200	C
250	FORTRAN

그림 7-5 Summer 테이블의 분해

수정된 (잘 설계된) 계절학기 수강 테이블

- 삭제 이상 : 없음
  - 200번 학생의 계절학기 수강신청을 취소(투플을 삭제)하더라도, C강좌의 수강료 정보는 삭제되지 않음

SummerPrice(class, price)

class	price
FORTRAN	20000
PASCAL	15000
C	10000

SummerEnroll(sid,class)

sid	class
100	FORTRAN
150	PASCAL
~~200~~	~~C~~
250	FORTRAN

수정된 (잘 설계된) 계절학기 수강 테이블

- 삽입 이상 : 없음
  - 계절학기에 자바 강좌를 개설(자바 강좌 투플을 삽입)할 때, 불필요한 가짜 학생 정보 SID를 강제로 넣을 필요 없음

SummerPrice(class, price)

class	price
FORTRAN	20000
PASCAL	15000
C	10000
JAVA	25000

SummerEnroll(sid,class)

sid	class
100	FORTRAN
150	PASCAL
200	C
250	FORTRAN

수정된 (잘 설계된) 계절학기 수강 테이블

- 수정 이상 : 없음
  - FORTRAN 강좌의 수강료를 20000원에서 15000원으로 수정하더라도, FORTRAN 강좌 수강료 데이터의 불일치가 발생하지 않음

SummerPrice(class, price)

class	price
FORTRAN	~~20000~~ 15000
PASCAL	15000
C	10000

SummerEnroll(sid,class)

sid	class
100	FORTRAN
150	PASCAL
200	C
250	FORTRAN

함수 종속성

함수 종속성(FD, Functional Dependency)

▣ 함수 종속성

- X와 Y를 임의의 속성들의 집합이라고 할 때, X의 값이 Y의 값을 유일하게(unique) 결정(X의 값에 대해 반드시 한 개의 Y값이 대응)한다면 "X는 Y를 함수적으로 결정한다(functionally determines)"라고 함
- X → Y로 표기하고, "Y는 X에 함수적으로 종속된다" 혹은 "X는 Y의 결정자(determinant)" 혹은 "X는 Y를 결정한다." 혹은 "Y는 X에 의해 결정된다" 라고 함
- 함수 종속성은 릴레이션의 가능한 모든 인스턴스에 대해 성립해야 함
- FD는 릴레이션 스키마에 대한 주장이지 릴레이션의 특정 인스턴스에 대한 주장이 아니다.

함수 종속성(FD, Functional Dependency)

학생수강성적

학생번호	학생이름	주소	학과	학과사무실	강좌이름	강의실	성적
501	박지성	영국 맨체스타	컴퓨터과	공학관101	데이터베이스	공학관 110	3.5
401	김연아	대한민국 서울	체육학과	체육관101	데이터베이스	공학관 110	4.0
402	장미란	대한민국 강원도	체육학과	체육관101	스포츠경영학	체육관 103	3.5
502	박지성	대한민국 서울	전자과	공학관101	자료구조	공학관 111	4.0
501	박지성	영국 맨체스타	컴퓨터과	공학관101	자료구조	공학관 111	3.5

❖ 함수종속관계 성립 예

학생번호 → 학생이름
학생번호 → 주소
강좌이름 → 강의실
학과 → 학과사무실

❖ 함수종속관계 성립하지 않는 예

학생이름 → 강좌이름
학과 → 학생번호
학생이름 → 학과
주소 → 학생번호

함수 종속성 다이어그램

▣ 함수 종속성 다이어그램(functional dependency diagram)은 함수 종속성을 나타내는 표기법

- 릴레이션의 속성 : 직사각형
- 속성 간의 함수 종속성 : 화살표
- 복합 속성 : 직사각형으로 묶어서 그림

그림 7-8 학생수강성적 릴레이션의 함수 종속성 다이어그램

함수 종속성과 키

- 릴레이션 R에서 속성 X가 후보 키/기본 키이면, R의 모든 속성 Y에 대해 X → Y 성립
  - 즉, 후보 키/기본 키는 릴레이션의 모든 속성에 대해 결정자임
- 그러나, 함수종속 X → Y의 경우
  속성 X가 반드시 후보 키/기본 키라는 것을 요건으로 하지 않음

함수 종속성 예제

- 릴레이션 R에서 아래의 함수 종속성이 성립하는가?

R

A	B	C
2	3	8
5	9	6
7	9	6
5	2	2

[함수 종속성]
① A → B
② B → C
③ (B, C) → A
④ (A, B) → C

정규형

개념

- 정규화(Normalization)
  - 주어진 릴레이션 스키마가 잘못되어 이상현상이 발생하는 릴레이션 스키마를 (함수적 종속성 등의 종속 이론을 이용하여) 분해하여, 이상 현상이 발생하지 않는 바람직한 릴레이션 스키마로 만들어 가는 과정
- 정규형(Normal Form)
  : 테이블의 정규화된 정도(등급)
  - 이상현상이 있는 릴레이션은 이상현상을 일으키는 함수 종속성의 유형에 따라 등급 구분
  - 릴레이션은 정규형 개념으로 구분하며, 정규형이 높을수록 이상현상은 줄어듦

개념

▣ 정규화(Normalization)의 원칙

정규화 = 스키마 변환 (S->S')

① 무손실 표현
- 같은 의미의 정보 유지
- 그러나 더 바람직한 구조

② 데이타의 중복성 감소

③ 릴레이션 분해
- 1개의 릴레이션을 여러 개의 릴레이션으로 표현
- 릴레이션 각각에 대해 독립적 조작이 가능

릴레이션 분해

▣ 릴레이션 R의 애트리뷰트가 A1 ... An이라고 하자. R의 분해란 R을 다음과 같이 둘 이상의 릴레이션으로 대체하는 것이다.
- ◆ 새 릴레이션들은 R의 애트리뷰트 중 일부씩만 가지고 (R에 없던 애트리뷰트는 갖지 않으며),
- ◆ R의 애트리뷰트들은 모두 새 릴레이션 중 어느 하나에는 나타난다.

▣ R을 분해한다는 것은 R의 인스턴스를 저장하는 대신, 분해로 생성된 릴레이션들의 인스턴스를 저장하겠다는 것이다.

▣ 예) R(S,N,L,R,W,H)를 R1(S,N,L,R,H)와 R2(R,W)로 분해 가능

▣ 꼭 필요할 때에만 분해하도록 한다.

릴레이션 분해 : 고려할 점

▣ 주어진 스키마에 문제가 있어서 나누기를 할 때, 다음 특성을 가져야 한다.

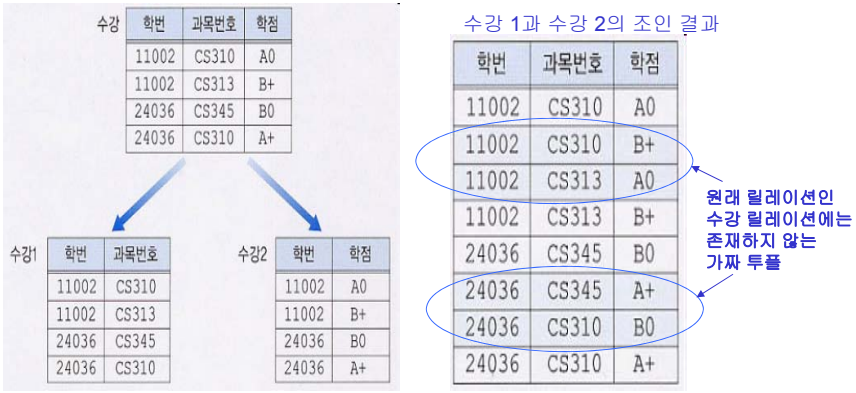
1) 가짜 투플을 만들지 않는 나누기
- 가짜 투플 : 릴레이션을 잘못 나누거나 새로 만든 작은 릴레이션을 다시 조인하였을 때 원래 릴레이션에는 없었던 투플

2) 함수 종속성을 유지하는나누기

3) 불필요하게 자료를 되풀이하지 않는 나누기

릴레이션 분해 : 고려할 점

▣ 예 : 가짜 투플이 만들어지는 잘못된 릴레이션 분해



릴레이션 분해 : 고려할 점

- 예 : 불필요하게 자료를 되풀이한 나누기

학생1

학번	이름	이메일
11002	이홍근	sea@hanmail.net
24036	김순미	smkim@venus.uos.ac.kr

학생2

학번	이름
11002	이홍근
24036	김순미

학생3

학번	이메일
11002	sea@hanmail.net
24036	smkim@venus.uos.ac.kr

릴레이션 분해 : 무손실-조인 분해 (Lossless Join Decompositions)

- FD집합 F를 만족하는 모든 인스턴스 r에 대해 다음이 성립하면, R을 X와 Y로 분해하는 것이 F에 대해 무손실-죠인(lossless-join)이라고 한다:
  - $\pi_X(R) \bowtie \pi_Y(R) = R$
- 중복성 제거를 위한 분해라면 반드시 무손실이어야 한다!
- 특히, 함수 종속 U->V가 R에 대해 성립할 때, R을 UV와 R-V로 분해하면 무손실-죠인 분해가 된다.
  - 릴레이션 R의 어트리뷰트들 (S,N,L,R,W,H)에 함수종속 관계로 S->SNLRWH 와 R->W가 있다고 한다면
    - R(S,N,L,R,W,H)를 R1(S,N,L,R,H)와 R2(R,W)로 분해
    - (함수적 종속관계를 이용하여) R2(R,W)를 새로 만들고 R1에서는 W를 뺌

정규형 종류

- 차수가 높을수록 데이터베이스의 무결성을 저해할 수 있는 논리적 모순이 존재할 가능성이 줄어들고, 중복된 데이터가 존재할 가능성도 줄어든다.

제1정규형 (1NF)

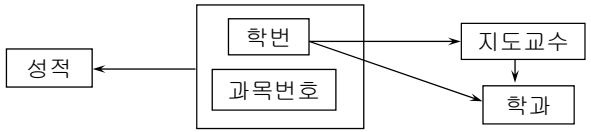
- 정의
  - 모든 속성값이 원자값(atomic value)만으로 된 릴레이션
  - 제 1 정규형은 릴레이션 정의의 일부분을 이루고 있으므로, 관계형 모델의 릴레이션이라면 모두 제 1정규형을 만족한다고 볼 수 있음
- 예 : 제 1 정규형을 만족하지 않는 릴레이션

고객취미들(이름, 취미들)

이름	취미들
김연아	인터넷
추신수	영화, 음악
박세리	음악, 쇼핑
장미란	음악
박지성	게임

제1정규형 (1NF)

- 예 : 수강지도 릴레이션(제 1정규형 만족)
  - 수강지도 (학번,지도교수,학과,과목번호,성적)
  - 기본키(후보키) : (학번,과목번호)
  - 함수 종속 : (학번,과목번호)->성적
    학번->지도교수
    학번-> 학과
    지도교수->학과



제1정규형 (1NF)

- 1NF에서의 이상 : 예

수강 지도

학번	지도교수	학과	과목번호	성적
100	P1	컴퓨터	C413	A
100	P1	컴퓨터	E412	A
200	P2	전기	C123	B
300	P3	컴퓨터	C312	A
300	P3	컴퓨터	C324	C
300	P3	컴퓨터	C413	A
400	P1	컴퓨터	C312	A
400	P1	컴퓨터	C324	A
400	P1	컴퓨터	C413	B
400	P1	컴퓨터	C412	C

제1정규형 (1NF)

- 1NF에서의 이상 : 예
  ① 삽입이상
  - 500번 학생의 지도교수가 P4라는 사실의 삽입
    : 어떤 교과목을 등록하지 않는 한 삽입 불가능

수강 지도

학번	지도교수	학과	과목번호	성적
100	P1	컴퓨터	C413	A
100	P1	컴퓨터	E412	A
200	P2	전기	C123	B
300	P3	컴퓨터	C312	A
300	P3	컴퓨터	C324	C
300	P3	컴퓨터	C413	A
400	P1	컴퓨터	C312	A
400	P1	컴퓨터	C324	A
400	P1	컴퓨터	C413	B
400	P1	컴퓨터	C412	C
500	P4		??	

제1정규형 (1NF)

- 1NF에서의 이상 : 예
  ② 삭제이상
  - 200번 학생이 C123의 등록을 취소하여 이 투플을 삭제할 경우 지도교수가 P2라는 정보까지 손실됨

수강 지도

학번	지도교수	학과	과목번호	성적
100	P1	컴퓨터	C413	A
100	P1	컴퓨터	E412	A
~~200~~	~~P2~~	~~전기~~	~~C123~~	~~B~~
300	P3	컴퓨터	C312	A
300	P3	컴퓨터	C324	C
300	P3	컴퓨터	C413	A
400	P1	컴퓨터	C312	A
400	P1	컴퓨터	C324	A
400	P1	컴퓨터	C413	B
400	P1	컴퓨터	C412	C

제1정규형 (1NF)

- 1NF에서의 이상 : 예

③ 수정이상

- 400번 학생의 지도교수를 P1에서 P3로 변경할 경우 학번이 400인 4개 튜플의 지도교수 값을 P3로 변경해야 함

수강 지도

학번	지도교수	학과	과목번호	성적
100	P1	컴퓨터	C413	A
100	P1	컴퓨터	E412	A
200	P2	전기	C123	B
300	P3	컴퓨터	C312	A
300	P3	컴퓨터	C324	C
300	P3	컴퓨터	C413	A
400	~~P1~~ P3	컴퓨터	C312	A
400	~~P1~~ P3	컴퓨터	C324	A
400	~~P1~~ P3	컴퓨터	C413	B
400	~~P1~~ P3	컴퓨터	C412	C

제1정규형 (1NF)

- 1NF에서의 이상 원인 : 예(수강 지도 릴레이션)
  - 학번,지도교수 데이터에 변경을 할 때, 문제 발생
    - 함수종속관계 '학번->지도교수'가 성립하기 때문에 학번, 지도교수 데이터에 대해서만 데이터 변경이 가능하지만,
    - 키인 (학번, 과목번호)가 있기 때문에, 학번, 지도교수 데이터만을 변경할 때 문제가 발생함

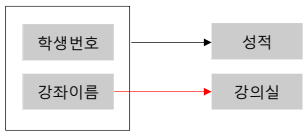
제1정규형 (1NF)

- 완전 함수 종속과 부분 함수 종속

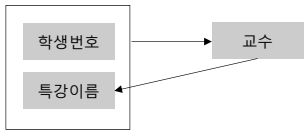
복합 속성 X에 대하여 X->Y가 성립할 때

- 완전 함수 종속 (FFD:Full Functional Dependency)
  - X' ⊂ X 이고 속성 X'이 결정자인 함수 종속 관계가 존재하지 않음
- 부분 함수 종속 (partial functional dependency)
  - X' ⊂ X 이고 속성 X'이 결정자인 함수 종속 관계가 존재함

- 예)



부분 함수 종속



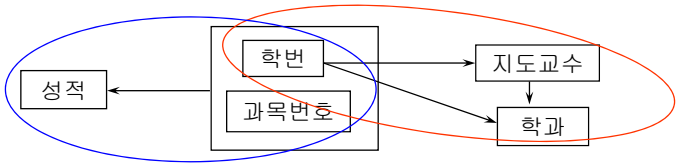
완전 함수 종속

제1정규형 (1NF)

- 1NF 이상의 원인
  - 키에 부분 함수 종속된 속성이 존재
- 1NF 이상의 해결
  - 릴레이션을 분해 (부분 함수 종속을 제거)

⇒ 2NF

- 예) 수강지도 (학번,지도교수,학과,과목번호,성적)

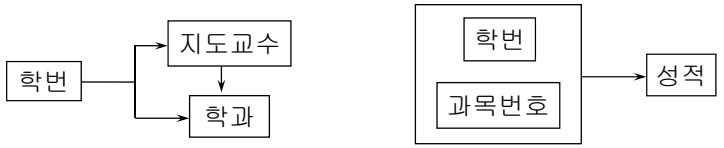


제2정규형 (2NF)

- 정의
  - 1NF이고, 어떤 키에도 속하지 않는 모든 속성들이 키에 완전 함수 종속
  - 키가 두 개 이상의 속성으로 구성되었을 경우에만 제1정규형이 제2정규형을 만족하는가를 고려할 필요가 있음(즉 키가 한 개의 속성으로 이루어진 릴레이션이 제1정규형을 만족하면 제2정규형도 만족함)

제2정규형 (2NF)

- 예 : 수강지도 ⇒ 지도, 수강 릴레이션
  - 지도 (학번,지도교수,학과)
    - 학번->지도교수
    - 학번->학과
    - 지도교수->학과
  - 수강 (학번,과목번호,성적)
    - (학번,과목번호)->성적



제2정규형 (2NF)

- 2NF에서의 이상 : 예

지도

학번	지도교수	학과
100	P1	컴퓨터
200	P2	전기
300	P3	컴퓨터
400	P1	컴퓨터

제2정규형 (2NF)

- 2NF에서의 이상 : 예
  - ① 삽입이상
    - 어떤 지도교수가 특정 학과에 속한다는 사실의 삽입 불가능

지도

학번	지도교수	학과
100	P1	컴퓨터
200	P2	전기
300	P3	컴퓨터
400	P1	컴퓨터
??	P4	기계

제2정규형 (2NF)

- 2NF에서의 이상 : 예
  - ② 삭제이상
    - 300번 학생의 투플을 삭제하면 지도교수 P3가 컴퓨터과에 속한다는 정보 손실

지도

학번	지도교수	학과
100	P1	컴퓨터
200	P2	전기
~~300~~	~~P3~~	~~컴퓨터~~
400	P1	컴퓨터

제2정규형 (2NF)

- 2NF에서의 이상 : 예
  - ③ 수정이상
    - 지도교수 P1의 소속이 컴퓨터과에서 전자과로 변경된다면 100과 400번 학생의 투플을 모두 변경하여야 함

지도

학번	지도교수	학과
100	P1	~~컴퓨터~~ 전자과
200	P2	전기
300	P3	컴퓨터
400	P1	~~컴퓨터~~ 전자과

제2정규형 (2NF)

- 2NF에서의 이상 원인 : 예(지도 릴레이션)
  - 지도교수,학과 데이터에 변경을 할 때, 문제 발생
    - 함수종속관계 '지도교수->학과'가 성립하기 때문에 지도교수, 학과 데이터에 대해서만 데이터 변경이 가능하지만,
    - 키인 (학번)이 있기 때문에, 지도교수, 학과 데이터만을 변경할 때 문제가 발생함

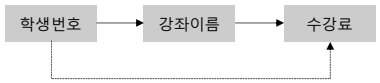
제2정규형 (2NF)

- 이행적 함수 종속 (TD, Transitive Dependency)
  - 한 릴레이션의 속성 A, B, C가 주어졌을 때 속성 C가 이행적으로 A에 종속한다(A→C)는 것의 필요 충분 조건은

    $A \rightarrow B \ \wedge \ B \rightarrow C$

    가 성립하는 것
  - A가 릴레이션의 키라면 키의 정의에 따라 A→B와 A→C가 성립한다. 만일 C가 A외에 B에도 함수적으로 종속한다면(B → C), C는 A에 직접 함수적으로 종속하면서 B를 거쳐서 A에 이행적으로 종속한다.

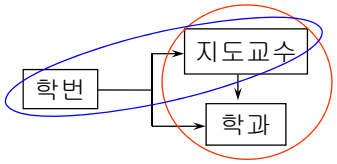
- 예) R(<u>학생번호</u>, 강좌이름, 수강료) : 이행적 함수 종속 발생



제2정규형 (2NF)

- 2NF 이상의 원인
  - 키가 아닌 속성들이 키에 이행적 함수 종속됨
- 2NF 이상의 해결
  - 릴레이션을 분해 (이행적 함수 종속을 제거)

    ⇒ 2NF
- 예) 지도 (학번,지도교수,학과)



제3정규형 (3NF)

- 정의(3NF)
  - 2NF이고, 키가 아닌 모든 속성들이 키에 이행적 함수 종속되지 않음 (2NF이고, 키가 아닌 모든 애트리뷰트가 키에 직접 종속)

제3정규형 (3NF)

- 예 : 지도 ⇒ 학생지도, 지도교수_학과 릴레이션
  - 학생지도 (학번,지도교수)

    학번->지도교수
  - 지도교수_학과 (지도교수,학과)

    지도교수 ->학과

학생지도

학번	지도교수
100	P1
200	P2
300	P3
400	P1

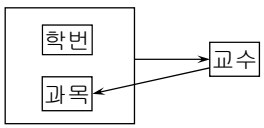
지도교수_학과

지도교수	학과
P1	컴퓨터
P2	전기
P3	컴퓨터

제3정규형 (3NF)

- 3NF에서의 이상 : 예
  - 수강과목 (학번,과목,교수)
    - 후보키 : (학번,과목), (학번,교수)
    - 기본키 : (학번,과목)
    - 함수종속 : (학번,과목)->교수

      교수 ->과목



수강과목

학번	과목	교수
100	프로그래밍	P1
100	자료구조	P2
200	프로그래밍	P1
200	자료구조	P3
300	자료구조	P3
300	프로그래밍	P4

제3정규형 (3NF)

- 3NF에서의 이상 : 예
  - ① 삽입이상
    - 교수 P5가 자료구조를 담당한다는 사실의 삽입은 수강 학생이 있어야 가능

수강과목

학번	과목	교수
100	프로그래밍	P1
100	자료구조	P2
200	프로그래밍	P1
200	자료구조	P3
300	자료구조	P3
300	프로그래밍	P4
??	자료구조	P5

제3정규형 (3NF)

- 3NF에서의 이상 : 예
  - ② 삭제이상
    - 100번 학생이 자료구조를 취소하여 투플을 삭제하면 P2라는 교수 정보도 함께 삭제됨

수강과목

학번	과목	교수
100	프로그래밍	P1
~~100~~	~~자료구조~~	~~P2~~
200	프로그래밍	P1
200	자료구조	P3
300	자료구조	P3
300	프로그래밍	P4

제3정규형 (3NF)

- 3NF에서의 이상 : 예
  - ③ 수정이상
    - P1이 프로그래밍 대신 자료구조를 담당하게 되면 P1이 나타난 모든 투플을 변경하여야 함

수강과목

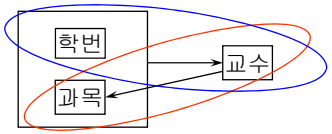
학번	과목	교수	
100	~~프로그래밍~~	P1	자료구조
100	자료구조	P2	
200	~~프로그래밍~~	P1	자료구조
200	자료구조	P3	
300	자료구조	P3	
300	프로그래밍	P4	

제3정규형 (3NF)

- 3NF에서의 이상 원인 : 예(수강과목 릴레이션)
  - 교수, 과목 데이터에 변경을 할 때, 문제 발생
    - 함수종속관계 '교수->과목'이 성립하기 때문에 교수, 과목 데이터에 대해서만 데이터 변경이 가능하지만,
    - 키인 (학번,과목)이 있기 때문에, 교수, 과목 데이터만을 변경할 때 문제가 발생함

제3정규형 (3NF)

- 3NF 이상의 원인
  - 교수가 결정자이나 키가 아님
- 3NF 이상의 해결
  - 릴레이션을 분해(모든 결정자가 키가 되도록 분해)
  - ⇒ BCNF
- 예) 수강과목 (학번,과목,교수)
  - 후보키 : (학번,과목), (학번,교수)



보이스/코드 정규형(BCNF)

- 정의
  - 3NF이고, 릴레이션 R의 모든 결정자가 키이면 릴레이션 R은 BCNF에 속한다.
- 릴레이션 R이 BCNF에 속하면 R은 제1, 제2, 제3 정규형에 속함
- 강한 제3정규형(strong 3NF)이라고도 함

보이스/코드 정규형(BCNF)

- 예(BCNF) : 수강과목 ⇒ 수강교수, 과목교수

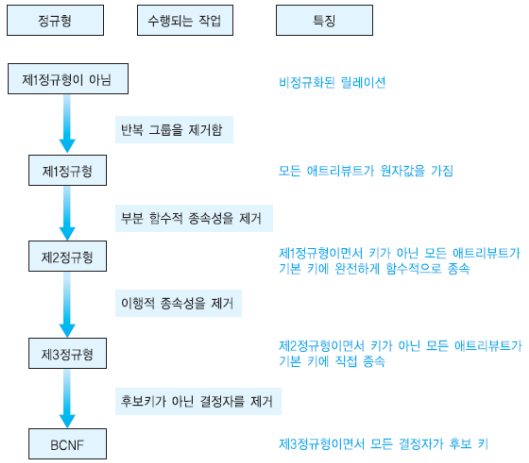
수강교수

학번	교수
100	P1
100	P2
200	P1
200	P3
300	P3
300	P4

과목교수

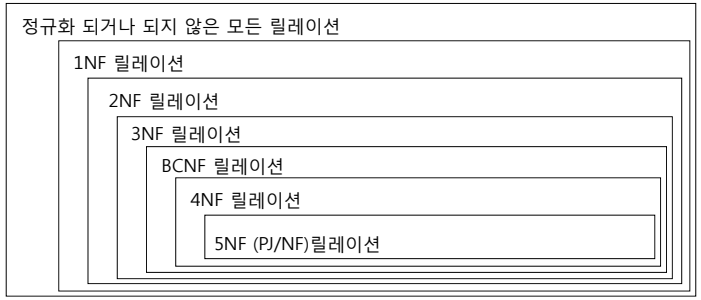
교수	과목
P1	프로그래밍
P2	자료구조
P3	자료구조
P4	프로그래밍

정규형 요약



정규형들 간의 포함 관계

- 대부분의 릴레이션은 BCNF까지 정규화하면 실제적인 이상 현상이 없어지기 때문에 보통 BCNF까지 정규화를 진행함



정규화 다시 보기

정규화의 의미

- 정규화의 개념
  - 자료의 손실이나 불필요한 정보의 도입 없이 데이터 일관성, 최소한의 데이터 중복, 최대한의 데이터 안전성을 확보하기 위한 구조로 전환하는 방법
  - 주어진 하나의 릴레이션을 이상 현상이 적게 발생하는 좋은 형태의 작은 릴레이션들로 분해하는 방법을 설명하는 이론
  - 정규화 이론은 이상 현상을 야기하는 속성간의 종속 관계를 감지하여, 이러한 종속관계가 사라지도록 릴레이션을 작은 여러 릴레이션으로 무손실 분해하여 종속 관계를 제거하고, 결과적으로 이상 현상을 제거
- 정규화의 필요성
  - 저장공간 최소화
  - 데이터 중복에 따른 데이터 불일치성 최소화
  - 데이터의 삽입, 수정, 삭제에 따른 이상현상 제거

정규화의 의미

- 정규화 절차의 핵심
  - 하나의 릴레이션에는 하나의 주제(theme)만이 포함되도록 함
    - 이상 현상은 한 릴레이션에 여러 개의 주제가 섞여 발생하는 것으로 볼 수 있음
  - 한 릴레이션에는 그 릴레이션의 키에 의해 결정되는 속성만을 포함시키도록 함
  - 이상 현상을 최소화

데이터베이스 설계 과정과 정규화

- 정규화 전의 릴레이션 R이란 무엇인가?
  - R은 ER 다이어그램으로부터 만들어진 테이블
- ER 다이어그램이 잘 만들어져 있었다면, ER 다이어그램에서 생성되는 테이블들에 대한 더 이상의 정규화는 필요 없음

=> 그렇다면, 어떻게 데이터베이스를 설계할까?

- ER 다이어그램을 만든 후 정규화가 잘 되어 있는지 여부를 확인하고, 정규화가 되어 있지 않다면 정규화를 추가 시행

비정규화(Denormalization)

- 정규화를 통해 릴레이션이 분해되면, 원래의 릴레이션에서 검색했을 때보다 검색 속도가 떨어질 수 있음
- 성능 개선을 목적으로 비정규화(역정규화, denormalizaton)를 하기도 한다.
- 예
  - 질의어> 고객번호,계좌번호,잔액을 보여라
    - 계좌(계좌번호, 잔액)
    - 예금(고객번호, 계좌번호)

    ⇒ '계좌'와 '예금'간의 조인 필요
  - 검색 속도가 매우 중요한 경우,
    - 예금_계좌(계좌번호, 잔액, 고객번호)로 정규화되지 않은 형태로 변경

정규화, 비정규화

- 정규화를 완전한 수준까지 한다고 해서 항상 좋은 데이터베이스 설계가 나오는 것은 아니다.
- 비정규화를 남용하는 것은 데이터베이스의 이상 현상이 많아지는 것을 의미한다.

⇒ 정규화와 비정규화의 정도는 데이터베이스 설계자의 경험과 적용되는 각 상황과 데이터의 특성에 따라 결정되어야 하는 어려운 문제임

⇒ 그러나 1차적으로는 정규화를 수행할 것. 성능문제가 발생하는 경우, 성능개선을 위한 다른 방법을 모색한 후 비정규화를 검토하는 것이 좋음